

# **README: Climate Data Analysis Tools (CDAT) version 4.0**



*CDAT Installation Help Guide*  
*April 1, 2005*

<http://www-pcmdi.llnl.gov/software/support.php>

## TABLE OF CONTENTS

<b>1</b>	<b><i>About This README – April 1, 2005.....</i></b>	<b><i>4</i></b>
<b>2</b>	<b><i>Introduction .....</i></b>	<b><i>4</i></b>
2.1	Project goal.....	4
2.2	Project overview.....	4
<b>3</b>	<b><i>Software Documentation .....</i></b>	<b><i>5</i></b>
<b>4</b>	<b><i>Software Platforms .....</i></b>	<b><i>5</i></b>
4.1	Supported platforms.....	5
4.2	Other platforms .....	5
<b>5</b>	<b><i>Install Environment.....</i></b>	<b><i>5</i></b>
5.1	Where should I install CDAT? .....	6
5.2	Platform compiler options.....	6
5.3	Python environment variables .....	6
<b>6</b>	<b><i>Express Install .....</i></b>	<b><i>7</i></b>
6.1	Express installation.....	7
6.2	Express installation help and options.....	7
6.3	CDMS only express installation instruction .....	7
6.4	Special MacIntosh OS X 10.3.x installation instructions.....	7
6.4.1	Fink CDAT install: .....	8
6.5	Special Cygwin (Windows XP) installation instructions .....	8
<b>7</b>	<b><i>Expert “Separate” Build Instructions.....</i></b>	<b><i>9</i></b>
7.1	PYSRC: Install Python (if necessary):.....	9
7.2	EXSRC: Install the “external” software (if necessary): .....	10
7.3	CDAT: Installing CDAT itself:.....	11
7.3.1	SETUP configuration file (optional):.....	11
7.3.2	Additional optional steps: .....	11
7.3.3	Running "install.py": .....	11
<b>8</b>	<b><i>Trouble Shooting the Build Process.....</i></b>	<b><i>12</i></b>
<b>9</b>	<b><i>Testing Software .....</i></b>	<b><i>12</i></b>
<b>10</b>	<b><i>Removing (or Cleaning) Software.....</i></b>	<b><i>13</i></b>
<b>11</b>	<b><i>Contribution (“contrib”) Packages .....</i></b>	<b><i>13</i></b>
<b>12</b>	<b><i>Special Notes on Specific Packages .....</i></b>	<b><i>14</i></b>
12.1	OPeNDAP (a.k.a., DODS) .....	14
12.2	Gplot .....	14
12.3	SCRIP interpolation .....	14
12.4	R statistical package .....	14
12.5	Scientific Python .....	15

<b>12.6</b>	<b>VTK package.....</b>	<b>15</b>
<b>13</b>	<b><i>Mailing List.....</i></b>	<b><i>15</i></b>
<b>14</b>	<b><i>Support and Training .....</i></b>	<b><i>15</i></b>
<b>15</b>	<b><i>A Special Thank You Note .....</i></b>	<b><i>16</i></b>

## 1 About This README – April 1, 2005

This is the Climate Data Analysis Tools (CDAT) version 4.0 source distribution. LLNL URL-CODE-2002-0021. (Please see file Legal.htm for our open-source license.)

CDAT and its documentation are freely available at <http://www.pcmdi.llnl.gov>, the home page of the Program for Climate Model Diagnosis and Intercomparison (PCMDI). (Please select the “SOFTWARE” menu at the top of this website).

For release notes, see the ReleaseNotes document.

The CDAT research and development team has prepared this README document. This document is aimed mainly at those wishing to install CDAT. The table of contents includes:

### Contents

1. About this README
2. Introduction
3. Software documentation
4. Software platforms
5. Installation environment variables and compiler options
6. Express installation
7. Expert “separate” build instructions
8. Trouble Shooting the Build Process
9. Testing software
10. Removing software
11. Contribution (“contrib”) packages
12. Special notes on specific packages
13. Mailing list
14. Support and training
15. Thank You

## 2 Introduction

### 2.1 Project goal

The primary goal of CDAT is to address the formidable challenges associated with analysis of simulations from global Earth System models. Through a combination of emerging community technologies, CDAT provides a seamless and powerful environment to enable the next phase of climate research.

### 2.2 Project overview

The Climate Data Analysis Tools (CDAT) is a software infrastructure that uses Python, an object-oriented scripting language, to link together separate software subsystems and packages, thus forming an integrated environment for solving model diagnosis problems. The power of this software system comes from Python and its ability to seamlessly interconnect software. Python provides a general purpose and full-featured scripting language with a variety of user interfaces including command-line interaction, stand-alone scripts (applications) and graphical user interfaces (GUI). The CDAT subsystems, implemented as Python modules, provide access to and management of gridded data (Climate Data Management System or CDMS); large-array numerical operations (Numerical Python); and visualization (Visualization and Control System or VCS).

CDAT is a freely available open source package, that is maintained, enhanced, and distributed freely by its user community. The home for CDAT sources and bug/patches can be downloaded from the PCMDI home page:

- <http://www.pcmdi.llnl.gov>

(Select the “SOFTWARE” menu, then the “About Software” menu item for more information on CDAT.)

### 3 Software Documentation

Extensive CDAT documentation can be found at:

- <http://www.pcmdi.llnl.gov/software/support.php>

New CDAT users should start with the “Climate Data Analysis Tools (CDAT): A Beginner’s Guide” document.

Online CDAT and VCDAT tutorials, examples, and additional module documentation can be found at the above URL.

For Python documentation and support, please visit the home page at: <http://www.python.org>.

### 4 Software Platforms

#### 4.1 Supported platforms

CDAT is fully supported on the following platforms:

- MacIntosh OS X 10.3.x/PowerPC
- RedHat Enterprise Linux WS 3.x/i386
- RedHat Fedora Core 1, 2, and 3/i386
- RedHat Linux 8.x and 9.x/i386
- SGI Altrix (64-bit) running RedHat Linux
- Sun/Solaris 8 and 9
- SuSE Linux 8.x and 9.x/i586

#### 4.2 Other platforms

CDAT is partially supported on the following additional platforms:

- Cygwin (Windows) 1.5.x/i386
- HP-UX 11
- IBM AIX 5L
- Linux flavors not mentioned above (e.g., Mandrake, Caldera, and Debian)
- OSF1 V4.x
- SGI IRIX 6.5

### 5 Install Environment

Software installation is divided into three parts: PYSRC, EXSRC, and CDAT. The PYSRC distribution contains everything needed to make a Tcl/Tk Python executable that is current enough to run CDAT. (**Note:** if you already have a suitable Python installation, the PYSRC installation can be skipped. See details below in section 7.) The EXSRC distribution contains required external packages necessary to support CDAT. The CDAT distribution contains Python

modules and subsystems that are explicitly designed to for the analysis of climate simulation and climate observed data.

## 5.1 Where should I install CDAT?

The notation <CDAT\_INSTALL\_DIRECTORY> indicates the full path name of the directory into which CDAT is to be installed.

The notation <CDAT\_SRC\_DIRECTORY> indicates the full path name of the directory where you uncompressed and un-tarred the CDAT source code.

It is required that the parent directory <CDAT\_INSTALL\_DIRECTORY> exists, and that you have write permission in it. For distributions that will be used by others, for example /usr/local/cdat-4.0 would be an appropriate name for the <CDAT\_INSTALL\_DIRECTORY>.

However, /usr/local is a poor choice, as chances of collisions with other installations and associated unintended consequences are too high and uninstalling becomes difficult. The <CDAT\_INSTALL\_DIRECTORY> will be created if necessary, with the following subdirectories: bin, DODS, lib, include, info, man, sample\_data.

## 5.2 Platform compiler options

Most platforms will build CDAT, as is with the GNU Compiler Collection (GCC) compiler, but on some platforms the environment variable “CC” may need to be set before building CDAT. See the table below for details.

Platform	Set CC environment Variable	Notes
AIX	cc_r	
Cygwin	gcc	Use default gcc
HP-UX 11	cc +Z	
IRIX 6.5	cc	Use configuration option “-c irix” to get n32 X11
Linux flavors	gcc	Use default gcc
Mac OS X	gcc	Use default gcc
OSF1	cc	
Solaris 8	cc -mt	
Solaris 9	gcc	Use default gcc

The way to set the environment variable CC depends on the shell used. For example:

- csh or tcsh: setenv CC value
- bash or sh: set CC=value; export CC

If the value is more than one word, double quotes must surround it. For example:

- setenv CC "cc -mt"

## 5.3 Python environment variables

On all platforms you must unset the following Python environment variables, if they are set:

- unset PYTHONPATH
- unset PYTHONHOME

## 6 Express Install

### 6.1 Express installation

Check for platform-specific instructions above, and set the environment variable CC if needed.

**(Note:** On some platforms, the CC environment variable must be set before building.)

Change the directory name to that where CDAT was un-tarred. At the prompt, type:

- `cd <CDAT_SRC_DIRECTORY>`
- `./express_install <CDAT_INSTALL_DIRECTORY> [options]`

### 6.2 Express installation help and options

To get installation help or to view installation [options], type:

- `cd <CDAT_SRC_DIRECTORY>`
- `./express_install --help`

or

- `more <CDAT_SRC_DIRECTORY>/HELP.txt`

or e-mail the CDAT discussion mailing list at:

- `cdat-discussion@list.sourceforge.net`

### 6.3 CDMS only express installation instruction

Check for platform-specific instructions above and set the environment variable CC if needed.

To install a minimal configuration supporting CDMS only:

- `cd <CDAT_SRC_DIRECTORY>`
- `./express_install <CDAT_INSTALL_DIRECTORY> --cdms-only [options]`

### 6.4 Special Macintosh OS X 10.3.x installation instructions

**Optional changing the shell mode (i.e., `bash`, `tcsh`, `csch`, etc.):** Change your shell preference environment by typing: "`chsh`" at the prompt. Then edit the file that appears. (Follow instructions in the file for editing.) You then must reboot your machine for the change to take effect.

**Setting 'root':** To become the "su" root user in an xterm window, you must first set one menu item. To start, you will need to select the "Finder" icon on your desktop. In the "Finder" window, select the following icons: "Applications", then "Utilities", then "NetInfo Manager". In the main menu of the "NetInfo Manager" window and under the "Security" menu, select the "Authenticate" menu item and follow the instructions.

**Note:** If you don't remember the root password, you can put in the 1<sup>st</sup> Mac OS X install disk and change the password when the prompt message states that you should do so.

**Xcode Tools Installation:** If the `-lcr1.o` or `bundle1.o` errors occur on compilation, then you need to reinstall the Xcode Tools! This is an unfortunate bug resulting from conflicting installers. The correct way to fix this problem is to install the full Xcode Tools 1.1 CD, which you can download from <http://connect.apple.com/>. (For testing the Mac OS X 10.3.x CDAT installation, we installed the latest Xcode Tools (version 1.5).)

You will also need the X11 libraries. You can only get this from the Xcode Tools 1.0 CD. From the CD, go to "Packages", then install "X11SDK.pkg". This will place the following libraries in your /usr/X11R6/lib directory:

libFS.a	libXTrap.a	libXi.a	libXt.a	libfntstubs.a
libGL.a	libXau.a	libXinerama.a	libXtst.a	libfontconfig.a
libGLU.a	libXaw.a	libXmu.a	libXv.a	libfontenc.a
libGLw.a	libXcursor.a	libXmuu.a	libXvMC.a	libfreetype.a
libICE.a	libXdmcp.a	libXp.a	libXxf86misc.a	liboldX.a
libOSMesa.a	libXext.a	libXpm.a	libXxf86vm.a	libpsres.a
libSM.a	libXfont.a	libXrandr.a	libdps.a	libxkbfile.a
libX11.a	libXfontcache.a	libXrender.a	libdpstk.a	libxkbui.a
libXRes.a	libXft.a	libXss.a	libexpat.a	

**G77 installation:** Go to the <<http://hpc.sourceforge.net/>> website and then get the g77v3.4-bin.tar.gz (Jaguar/Panther) binary. Place the tar file in the "/" directory. Become root and issue the command: tar xvfz g77v3.4-bin.tar.gz. This will put the binaries in /usr/local/bin. (The g77 installation is only needed for some "contrib" modules and is not necessary for CDAT building.)

**CDAT installation:** Make sure your xterm window has g77 by typing: "which g77". If not found, then follow appropriate the steps above.

Next follow the directions in the "Express installation" section (i.e., section 6.1) to build and install CDAT.

#### 6.4.1 Fink CDAT install:

From the [Fink](http://fink.sourceforge.net/) website, "Fink is defined as a project that wants to bring the full world of Unix Open Source software to Darwin and Mac OS X. It has two main goals:

- First, to modify existing Open Source software so that it will compile and run on Mac OS X.
- Second, to make the results available to casual users as a coherent, comfortable distribution that matches what Linux users are used to.

The project offers precompiled binary packages as well as a fully automated build-from-source system."

To install "fink", follow the instructions located at the following website:

- (<http://fink.sourceforge.net/>).

By the courtesy of Jeff Whitaker, CDAT can be installed with the Fink application by typing:

- fink install cdat

Other useful Linux/Unix applications can be installed by using Fink. For a full listing of applications, type:

- fink list

## 6.5 Special Cygwin (Windows XP) installation instructions

Install Cygwin from <http://cygwin.com>. When selecting the Packages for the Cygwin installation, it is important to select "Install" for "Base" and "X11". This will ensure that all the appropriate libraries needed for CDAT builds properly. In general, it is not a bad idea to select "Install" for "All" packages if you have the disk space.

**Note:** The "Cygwin Setup" can be a little confusing for first time users. When the package selection window appears, you will see the package name, a cycle icon, and the text "Default". If you select the text "Default" it will change to "Install". At the very top of this



window you will see the “All” line. For the “All” line, we recommend that you toggle the selection from “Default” to “Install”. This toggle from “Default” to “Install” will also change all subsequent package preinstall indications from “Default” to “Install” and will insure that the packages needed for CDAT will be installed accordingly.

Once you’ve installed Cygwin, select the Cygwin icon (located on your desktop) or from the “start” menu select “All Programs” then “Cygwin Bash Shell”. In the window that appears, type:

- rebaseall

If your TMP or TEMP environment variables are not set properly, you may need to issue the following command:

- TMP=/tmp rebaseall

After issuing the “rebaseall” command, type:

- startx

**Note:** It is extremely important to issue the “rebaseall” command; otherwise threading for Python and CDAT will not work properly.

**Note:** If your \$HOME path has a space (or spaces) in it, then you will need to remove the space(s) by editing the “/etc/passwd” file. Make sure your account has “Computer Administrator” privileges. If not, then you will need to log on with an account that does. In the “/etc/passwd” file, find your user account and change your home directory path to remove all spaces. For example, if your path has:

- “/home/firstname lastname:/bin/bash”;

change it to:

- “/home/lastname:/bin/tcsh”.

When you have finished making your changes, you are required to reboot your machine for the changes to occur.

## 7 Expert “Separate” Build Instructions

Check for platform-specific instructions above and set the environment variable CC if needed.

(**Note:** On some platforms you may need to set the CC environment variable before building.)

CDAT can be built in three separate phases:

1. **PYSRC** - To install Python with tcl/tk:
  - cd <CDAT\_SRC\_DIRECTORY>/pysrc
  - ./install\_script <CDAT\_INSTALL\_DIRECTORY> (To skip this step, see PYSRC below)
2. **EXSRC** - To install the external software, CDAT requires:
  - cd <CDAT\_SRC\_DIRECTORY>/exsrc
  - ./install\_script <CDAT\_INSTALL\_DIRECTORY> (infrequently changing components)
3. **CDAT** - To install CDAT itself:
  - <CDAT\_INSTALL\_DIRECTORY>/bin/python install.py [options] (installs CDAT and user-contributed packages (contrib can be turned off with the option “--without-contrib”))

### 7.1 PYSRC: Install Python (if necessary):

The “pysrc” distribution contains everything you need to run CDAT: zlib, readline,

tcl, tk, and Python.

The source code versions distributed are:

- Python 2.4
- Tcl/Tk 8.4.9
- readline 4.3
- zlib 1.1.4

Reasons you might build a new Python include:

1. You do not have Python installed in a directory where you have write permission.
2. You get a message during CDAT installation saying that your Python is too old.
3. You do not want CDAT installed into your existing Python.
4. You want to avoid issues such as having the right Python but the wrong Tcl/Tk.

Everything needed for Python, including Tcl and Tk, are included in this package. If in doubt, please compare the versions of the tar files in <CDAT\_SRC\_DIRECTORY>/pysrc/src to be sure your version numbers are as recent as ours.

Obtain the pysrc distribution and unpack it. In subdirectory <CDAT\_SRC\_DIRECTORY>/pysrc, type:

- ./install\_script [--debug] <CDAT\_INSTALL\_DIRECTORY>

The build takes place in subdirectory “build”, and you'll find log files there.

You can clean up the build directories by issuing the command:

- cd <CDAT\_SRC\_DIRECTORY>/pysrc
- ./clean\_script

## 7.2 EXSRC: Install the “external” software (if necessary):

Our "exsrc" distribution contains additional required packages. These packages, except for Pyfort, were not written at PCMDI. We have separated them into their own distribution to reduce the amount of software the users must download and rebuild when new versions of CDAT are released and these parts haven't changed.

To build exsrc, type:

- cd <CDAT\_SRC\_DIRECTORY>/exsrc
- ./install\_script [--debug] <CDAT\_INSTALL\_DIRECTORY>

The build takes place in subdirectory “build,” and you'll find log files there.

You can clean up the build directories by issuing the command:

- cd <CDAT\_SRC\_DIRECTORY>/exsrc
- ./clean\_script

You only need to do this step once. Minor CDAT updates will not require you to download and build this part again.

If you need to rebuild only some parts of it, you can enable/disable any PACKAGE by adding the --with-PACKAGE or --without-PACKAGE, or build a single PACKAGE by using --PACKAGE-only. To view the list of external software packages, type:

- cd <CDAT\_SRC\_DIRECTORY>/exsrc
- ./install\_script . --help

The Packages available are:

- NetCDF 3.5.1
- DODS 3.4.3 (DODS default builds for Linux and Mac OS X 10.3.x. Porting DODS to other platforms such as Solaris and Cygwin are underway.)
- Pyfort 8.5.1
- Numeric 23.1
- XGKS 2.6.1
- Pmw 1.2
- Ghostscript 8.50 with jpeg 6b and libpng 1.2.8
- gplot 1989
- gifsicle 1.35
- pbmplus 1991 (Distributed but not used for Linux/Mac)
- netpbm 10.20 ( Replaces pbmplus on Linux and Mac system)
- gifmerge 1999
- R 2.0.1 (not installed by default, you'll need to pass --with-R)
- VTK 4.2.3 (not installed by default, you'll need to pass --with-VTK)
- cmake 1.8.3 (installed by VTK)

### 7.3 CDAT: Installing CDAT itself:

#### 7.3.1 *SETUP configuration file (optional):*

CDAT installation can be controlled with a configuration file. The installation begins by reading the file `installation/standard.py`. Options you can set are detailed in that file.

You can use an alternate configuration file (say `myconfig.py`) by using in step 1:

- `--configuration=myconfig`

Example of a configuration file: The file `installation/pcmdi.py` is an example of a configuration file, in this case the configuration we use at PCMDI. Once a configuration has been specified, it will not be redone in subsequent installs unless you use the “`--force`” option.

#### 7.3.2 *Additional optional steps:*

**Note** that in creating your own configuration file, you can use the full power of Python.

A “control file” also governs the install process. The standard control file is `installation/control.py`; `installation/debug.py` is used for the debug builds. You shouldn't have to change anything in these files.

#### 7.3.3 *Running "install.py":*

```
<CDAT_INSTALL_DIRECTORY>/bin/python install.py [options] [control-files]
```

In the following step you must execute the Python from which you intend to install CDAT. Either use a full path name or make sure that the desired python is the one being chosen by your path. To emphasize this we'll show the command as a full pathname below. Please read notes below before building.

A full usage description is given in file `HELP.txt`. It can also be viewed with

- `<CDAT_INSTALL_DIRECTORY>/bin/python install.py --help`.

Standard install (now includes “contrib” Packages):

- `<CDAT_INSTALL_DIRECTORY>/bin/python install.py`

Standard without contributed packages:

```
<CDAT_INSTALL_DIRECTORY>/bin/python install.py --without-contrib
```

To install after changing the configuration, downloading a new version, or changing platforms, use the `--force` option. For example:

- `<CDAT_INSTALL_DIRECTORY>/bin/python install.py --force`

To install using your own configuration file:

- `<CDAT_INSTALL_DIRECTORY>/bin/python install.py --configuration=myconfig`, where `myconfig` is your configuration file name.

A list of all packages installed will appear. If a package fails, an error message will be generated at the end of the install. (All failed Packages will be listed.)

If the build has failed for some packages, you should then examine the log file for the software component it was working on when it failed. For comparison, the subdirectory "logs/samples" contains the log files for a build on all platforms.

A certain class of errors, such as a failure to locate the X11 libraries, can result in the build halting in an inconsistent state. Typically you will fix such a problem by adding a configuration or control file and rebuilding. Be sure to use the `--force` option when you rebuild to insure a complete rebuild.

The set of packages to build is determined by the "packages." To modify the list of packages, make a control file that specifies the desired variable list.

Many of the contributed packages in subdirectory "contrib" require a Fortran compiler. If the default compiler chosen by Pyfort is not what you want, you may need to edit file `configuration.py` in the Pyfort source and reinstall it. See <http://pyfortran.sf.net>.

## 8 Trouble Shooting the Build Process

If problems occur when building any part of the CDAT component software, you can view the build error of an individual package in its log file, which is located in the directory:

- `<CDAT_SRC_DIRECTORY>/logs`

You can also find log files in the "build" subdirectories of "pysrc" and "exsrc". That is, in

- `<CDAT_SRC_DIRECTORY>/pysrc/build`
- `<CDAT_SRC_DIRECTORY>/exsrc/build`

Example logs can be found in the following directory:

- `<CDAT_SRC_DIRECTORY>/logs/samples`

## 9 Testing Software

To quickly test CDAT, type the following:

- `cd <CDAT_SRC_DIRECTORY>`
- `./test_script [-v] <CDAT_INSTALL_DIRECTORY>`

The `test_script` will run a basic set of `cdms` tests and start up `vc dat` for testing.

If `<CDAT_INSTALL_DIRECTORY>` is not given, then `test_script` will run the Python in your path. If the `-v` option is given, then it will not start up `VCDAT` (i.e., the GUI interface to CDAT).

**Note:** Test 6, the xml test, often prints out differences that are matters of form rather than substance. These occur, for example, if you used a different version of Python than we did. We are working to improve this test.

## 10 Removing (or Cleaning) Software

For building again on a different system, or after downloading a new version, or just to save disk space, issue the following commands:

- `cd <CDAT_SRC_DIRECTORY>`
- `./clean_script all`

To clean the separate build sections for PYSRC, EXSRC, and CDAT, issue the following commands:

- `cd <CDAT_SRC_DIRECTORY>/pysrc`
- `./clean_script`
- `cd <CDAT_SRC_DIRECTORY>/exsrc`
- `./clean_script`
- `cd <CDAT_SRC_DIRECTORY>`
- `./clean_script`

**Note:** To rebuild CDAT, use the “--force” option. If you wish to rebuild CDAT and restart over, it is best to remove the `<CDAT_INSTALL_DIRECTORY>`. We recommend the following steps:

- `cd <CDAT_SRC_DIRECTORY>`
- `./clean_script all`
- `rm -rf <CDAT_INSTALL_DIRECTORY>`
- `./express_install <CDAT_INSTALL_DIRECTORY> [options]`

## 11 Contribution (“contrib”) Packages

Below is the list of distributed “contrib” packages, with documentation provided in the source files (or online).

asciidata	
binaryio	
ComparisonStatistics	1.2
cssgrid	
dsgrid	
eof	
f2py	2.37.233-1545
grads	
laGraph	0.3.0.1
lmoments	
matplotlib	0.54
natgrid	
ort	
pyclimate	1.2.1
pygmt	2004/05/03
pyncl	1.2
PyX	0.6.3
regridpack	
Rpy	0.3.5
shgrid	

Scientific Python	2.2
spherepack	
Thermo	1.0
trends	

## 12 Special Notes on Specific Packages

### 12.1 OPeNDAP (a.k.a., DODS)

CDAT is automatically built with OPeNDAP (a.k.a., DODS) support on Linux and Mac OS X 10.3.x platforms. You must provide the binaries for the DODS libraries on all other platforms. (Make sure they've been compiled with the same compiler used to build CDAT.) To build with OPeNDAP client support, use the `--dods=/my_dods_dir` command option. (We recommend the use of the `--force` option as well).

If you do not want OPeNDAP support (e.g., for Linux and Mac OS X platforms) add the `--without-DODS` option to the build command line.

### 12.2 Gplot

The gplot utility is required to produce postscript files and is distributed within the `exsrc` directory. (See section 7.2 above.) It is installed automatically on all supported platforms. However if it fails, follow the instructions below. The built gplot executable is located in the `<CDAT_INSTALL_DIRECTORY>/bin` directory.

Instructions for build gplot are as follows:

- `cd <CDAT_SRC_DIRECTORY>/exsrc/src`
- `gunzip gplot.tar.gz ; tar xvf gplot.tar`
- `cd gplot`
- `make -f Makefile.your_platform gplot`

Choose an appropriate Makefile in subdirectory `gplot`. Modification of the Makefile may be required for finding X11.

You then can install the executable "gplot" anywhere in your execution path. For example,

- `cp -p gplot <CDAT_INSTALL_DIRECTORY>/bin/gplot`

### 12.3 SCRIP interpolation

CDAT Version 4 includes support for the SCRIP interpolation package developed at Los Alamos National Laboratory. SCRIP interpolates gridded data, and can be used with nonrectangular grids introduced in CDAT V4.

Because this package is standalone and is written in Fortran 90, it is not built by default.

**Note:** CDAT has a built-in regridder for rectangular grids. If you need the richer functionality of SCRIP, this package is included in "`<CDAT_SRC_DIRECTORY>/exsrc/src`" directory. See the SCRIP user guide for installation instructions.

### 12.4 R statistical package

The R statistical package is extremely long to build, but builds smoothly on Linux and Mac OS X 10.3.x platforms. Because of the time R takes to build, we've decided to remove it from the default build process. To include the R build, add the CDAT command line option:

- `cd <CDAT_SRC_DIRECTORY>/exsrc`

- `./install_script --with-R`
- `cd ../contrib/Rpy`
- `./install_script <CDAT_INSTALL_DIRECTORY>`

If you already have R built on your system, and wish to only build the Rpy contrib package, then make sure your `LD_LIBRARY_PATH` includes the path to your R distribution, and then run the following build line command:

- `cd <CDAT_SRC_DIRECTORY>`
- `<CDAT_INSTALL_DIRECTORY>/bin/python install.py --with-R`

## 12.5 Scientific Python

If OPeNDAP (a.k.a., DODS) is enabled, then SP will have DODS support.

## 12.6 VTK package

The VTK package is EXTREMELY long to build (about 45mn on some of the fastest Linux/PCs). Therefore, it is not built by default. To include the VTK build, add the CDAT command line option:

- `cd <CDAT_SRC_DIRECTORY>/exsrc`
- `./install_script --with-VTK`

For example,

- `cd <CDAT_SRC_DIRECTORY>`
- `./express_install --with-VTK`

or

- `cd <CDAT_SRC_DIRECTORY>/exsrc`
- `<CDAT_INSTALL_DIRECTORY>/bin/python install.py --with-VTK`

**Note:** VTK has been tested on RedHat Linux systems only. On some platforms you might need to set your `LD_LIBRARY_PATH` to point to the CDAT installation of Tcl/Tk.

## 13 Mailing List

There is a CDAT mailing list that allows users/developers to exchange ideas, information, tips, and discuss development issues. It is also intended for helping new CDAT users who need general questions answered. SourceForge hosts the CDAT mailing list. For more information and to subscribe, please visit the URL:

- <http://www.pcmdi.llnl.gov/software/support.php>

At this same URL, you can also access or view or report bugs, and make CDAT contributions.

## 14 Support and Training

As with every community software product, we are very interested in your comments, suggestions for improvements, work-arounds, and bug reports. The bug report database is also hosted by SourceForge and can be found at the URL:

- <http://www.pcmdi.llnl.gov/software/support.php>

The CDAT community is growing rapidly and becoming much too large for individual user support. Fortunately, there are a lot of really savvy CDAT users who are willing to help others and/or willing to contribute to the development and growth of CDAT. Please visit the URL above to find additional support documentation, tutorials on CDAT, and FAQs.

**Special Note:** On occasions, members of the core CDAT development team present tutorials on the installation, use, and development of CDAT.

## 15 A Special Thank You Note

At this time, we would like to express a special thanks to all the CDAT community supporters for the helpful suggestions and bug reports received throughout the years. CDAT has greatly improved from your suggestions, comments and patients. Thanks for your continual support and help in the development of the CDAT community software product.